A Technological Survey On Apache Spark And Hadoop Technologies.

Dr MD NADEEM AHMED, AASIF AFTAB, MOHAMMAD MAZHAR NEZAMI

Abstract: These days, whether it is mid-level or multilevel organizations alike accumulate enormous volume of data, and the only intension collecting these data is to: extract meaningful data called value through advanced level of data mining or analytics, and apply in decision making by personalized advertisement targeting , making the huge profit in business and extracting the rapidly using big data technologies. Big data due to its several features like value, volume, velocity, Variability and variety put further numerous challenges In this paper, we have completed an investigation of different huge information related advancements and we look at the advantages and disadvantages of these big data technologies and a comparative study of different proposed authors works over performance optimization of Big data related technologies has been done.

Index Terms: Hadoop, Apache Spark, Survey, Performance, HDFS,mapReduce,Bigdata

1 INTRODUCTION

The Total data up to 90's century is now today's sample data. According to Eric Schmidt, down of civilization till 2003 there was five (5) Exabyte of data/information but now that amount of data/information is created only in two (2) days because data/information is growing much faster than we expected. The reason of grow out of these data/information is that it is coming from every corner of the whole world. Twitter process 340 million messages weekly. Data generation in the last one year is equivalent to data generated in last 15 year. Facebook user generates 2.7 billion comments and likes. Amazon S3 storage adds more than one billion objects biweekly. E bay stores 90 petabytes of data about customer transactions. Enterprise information measures is no more in Tera bytes and Peta bytes but Exa and Zeta bytes. Because of progressive rise of data, it is very important how to fastly retrieve information from Big Data in the research institutes and enterprise. Presently, the system of Hadoop ecosystem has been more largely accepted by scientist. This ecosystem fuse HDFS, Map Reduce, Hive, HBase and Pig and so on. Pig and Hive are called batch Processing. Big Data—the term portraying the collection of new information from sources, for example, online individual movement, business exchanges, and sensor systems—it contains numerous trademark, for example, data as high speed, high volume, and high-assortment. For raising the decision making and perceptivity, this information high speed, high volume and high-assortment resources of data demanded imaginative structure and savvy information preparing according to BD's definition of Gartner. (gartner.com, 2013). For big data processing MapReduce [15] has become a recognized technology Since introduced by Google in 2004. Hadoop is an open-source usage of MapReduce. In different data analytic use cases it has been applied such as reporting, OLAP and web data search machine learning, data mining, social networking analysis and retrieval of information. To magnify and remove the disadvantages of Map Reduce Apache Spark was developed. Apache Spark shall be considered as advancement of Map Reduce. Apache Spark can process data 10x occasions quicker than guide Reduce on Disk and 100x occasions quicker than Map Reduce on memory. This can be achieved by minimizing the number of reading/compose activities to plate. It stores the moderate handling information in memory

.

Big data compromise of three broad components: -

Data	Storage & Retrieval, Search, Processing & Integration, Security
Analytics	Business Intelligence, reporting, Visualization, Statistical Modeling, Data mining & Predictive Modeling & machine Learning
Delivery & Consumption	Implementation of Analysis solution to the operational decision platform, Interactivity, customization and real time

2 LITERATURE SURVEY

In 2012, Floratou et al. [6] took a gander at Hive versus a similar database from Microsoft - SQL Server applying TPC-H specification. The outcomes demonstrate shows that at the four scale factors SQL Server is continually speedier than Hive for all TPC-H tests. Although when the dataset is litter the normal speedup of SQL Server over Hive is greater. Floratou et al. [7] in 2014, did another test contemplate: differentiating Hive against Impala applying a TPC-H like specification and two TPC-DS revived outstanding tasks at hand. The outcome exhibited that 2.1X to 2.8X speedier than Hive on Tez (Hive-Tez) for the TPC-H tests and Impala is 3.3X to 4.4X faster than Hive on MapReduce (Hive-MR).

- *Ph.D. (CS).*
- *Lecturer (CS)*
- *Lecturer, Department of Computer Sc.*
- *IFTM University, India*
- *College of CS/IT*
- *College of science and Arts Balqarn*
- *mdnadeemahmed.86@gmail.com*
- *Jazan University*
- *University of Bisha, Saudia Arabia*

- *aaftab@jazanu.edu.sa*
- *ndhami@ub.edu.sa*

In [8] , Yongqiang He , Rubao Lee , Yin Huai , Zheng Shao , Jain, N , Xiaodong Zhang , Zhiwei Xu has made RCFile to achieve brisk data stacking, speedy request planning and exceedingly capable limit space utilize .This record structure has good position of level line vertical section store structure and store structure. We will take this record association to test three-type request devices. In [9],they thought about some high effective circulated parallel databases and Hive ,and tune up the performance with utilizing several framework parameters gave by Hive, for example, HDFS Block Size ,Parallel Processing Slot Number and Partitions. We can get from their framework for presenting distinctive request on comprehensively appreciate the property. Jingmin Li In [10] composed the ongoing information investigation framework in light of the Impala. They clarified the purpose why Impala has been chosen by them by looking at the Hive. Impala inquiry proficiency is around 2~3 times than Hive. In [11], Lei Gu considered the Hadoop and Spark. They establish that regardless of the way that Spark is all things considered speedier than Hadoop in iterative sets of operation, it needs to bear for additional memory usage. The speed of the Spark advantage is crippled precisely when the memory isn't adequately sufficient to store as of late created direct results. So we should consider the execution of the memory, when we took a gander at three-type question devices.

S. No.	Paper	Author	Advantages	Issues
1	Performance Comparison of Hive & Impala and Spark SQL	Xiaopeng Li, Wenli Zhou	Note the similarity or dissimilarity between three-type question apparatuses in different file format effect on the memory and CPU, lastly, we observe of the document design for the inquiry time, talk about that the query speed of Impala, Parquet record group is taken made by Spark SQL is the quickest	Intended to upgrade SQL in the three kind inquiry apparatuses and look at the distinction when advancement. Additionally, need to examine other file format and alternative techniques of compression.
2	The Performance of SQL-on-Hadoop Systems: An Experimental Study	Xiongpai Qin, Yueguo Chen*, Jun Chen, Shuai Li, Jiesi Liu, Huijie Zhang	Here based on the TPCH benchmark author compares the execution of three agent SQL-on-Hadoop frameworks. Impala performs much better than Hive and Spark. Performance of SQL-on-Hadoop systems remarkably increased by using pipeline way of querying.	The execution of SQL-on-Hadoop frameworks can be additionally improved by applying further developed parallel database systems.
3	Performance issue and Query Optimization in Big Multidimensional Data	Jay Kiruthika, Dr Souheli Khaddaj	Gesture based communication and gamification of many application has led to increase in 3D storage. Here author compares the cost of 3D storage and Time execution.	More performance can be achieved by isolating executing time of the gadgets (client cost of the program)from the SQL query, experimenting on large 3 D databases involved complex structure will produce more results to work on
4	Performance Prediction for Apache Spark Platform	Kewen Wang, Mohammad Maifi Hasan Khan	Here authors introduced an exhibition expectation system Which runs job on Apache Spark platform, Author demonstrate models for assessing the execution of occupation by mirror the execution of genuine employment on a little scale on a continuous bunch . For execution of time and memory prophecy precision is observed to be high, For different applications the I/O cost prediction shows variation.	Need to check the the I/O cost expectation for an alternate arrangement of utilization. This may be happening because of it is unable to track network action in enough subtleties in a little scale impersonation.
5	Cross-Platform Resource Scheduling for Spark and MapReduce	Dazhao Cheng, Xiaobo Zhou, Palden Lama, Jun Wu, and Changjun Jiang	Author has noticed that if in YARN clusters, by running Spark and MapReduce causes noteworthy	While deploying more processing paradigms need to explore more

	on YARN		execution debasement in light of the fact that to the semantic gap between the dynamic application demands and the reservation-based resource allocation scheme of YARN. Therefore, a cross-platform resource scheduling middleware has been developed and designed , iKayak, that focused to enhance the utilization of cluster resource and for Spark-on-YARN deployment enhance application performance.	cross-platform resource scheduling proposal (e.g. Stormand Hive, Pig ,Shark, ,) on Hadoop YARN
6	An optimal approach for social data analysis in Big Data.	Ms, Kamala,V,R, Ms .L.MaryGladence	Authors implementation carried out by using Spark which is a faster data processing engine when compared to MapReduce and the code of lines used for the implementation is much lesser which increases simple and easy code maintainability.	By using concept of Spark Streaming, the import of data and processing of data shall be integrated. It gives resilient ,high-throughput, and scalable processing of data stream.
7	A Performance Study of Big Data Analytics Platforms	Pouria Pirzadeh, Michael Carey, Till Westmann Couchbase	Authors showes how a nested schema or the optimized columnar storage formats (Parquet and ORC) can ehnance performance in several cases. TPC-H benchmark has been used by author to assess four Big Data stages: Spark SQL AsterixDB, System-X (a parallel business RDBMS). also, Hive	Need to check why results demonstrated that no capacity organization, framework or construction variation given the best execution for the majority of the inquiries
8	Efficient Distributed Smith-Waterman Algorithm Based on Apache Spark	Bo Xu, Changlong Li, Hang Zhuang, Jiali Wang, Qingfeng Wang, Xuehai Zhou	CloudSW is a systematic Spark-based conveyed, Smith Waterman calculation improved for delivering the between pairwise groupings and getting the most K homogeneous sets of the arrangements in an on a level plane adaptable appropriated condition. CloudSW approve clients to get information from a few information sources, gives particular Techniques of activity, ASM and ATM, and furthermore allows clients to utilize distinctive designs.	Need to investigate different advances to update execution in extra hub bunch, and enhance for several category of sequence data.

3 BIGDATA PROPERTIES

Volume: Volume is likely the best known typical for enormous data; this is not all that much, considering more than 90 percent of each one of the present data was made in the current years. The present measure of data can truly be exceptionally shocking. Here are a couple of delineations, around 300 hours of video are exchanged to YouTube reliably. A normal almost around 1.1 trillion photos were taken in 2016, and that number is foreseen to rise by 9 percent in 2017. Users over an internet produces around 2.5 quintillion bytes of data per day[16].

Velocity: Speed alludes to the speed at which information is being produced, delivered, made, or revived. Of course, it sounds noteworthy that Facebook's information discount stores upwards of 300 petabytes of information, yet the speed at which new information is made ought to be considered. Facebook claims 600 terabytes of approaching information every day. Google alone procedures by and large more than "40,000 hunt queries each second," which generally means more than 3.5 billion inquiries for every day. As per survey [16] every person will generate 1.7 megabytes in just a second by 2020.

Variety: Concerning enormous information, we don't just need to oversee formed information yet what's more semi-organized and by and large unstructured data as well. As you can discover from the above delineations, most gigantic data is all in all unstructured, however by sound, picture, video archives, online person to person communication invigorates, and other substance courses of action there are also log records, click data, machine and sensor data, et cetera. Veracity: This is one of the dreadful characteristics of tremendous data. As any or most of the above properties increase, the veracity (conviction or on the other hand trust in the information) drops. This looks like, yet not the same as, realness or flimsiness (see underneath). Veracity implies more to the provenance or constancy of the data source, its circumstance, and that it is so critical to the examination in light of it. Validity: Like veracity, authenticity implies how exact and overhaul the data is for its arranged use. As per Forbes, a normal 60 percent of an information researcher's chance is spent purifying their data before having the ability to do any examination. The advantage from enormous information examination is simply in the same class as its hidden information, so you need to get incredible data organization practices to ensure unsurprising data quality, essential definitions, and metadataVolatility: How old does your data ought to be before it is seen as unessential, noteworthy, or not important anything else, To what degree does data ought to be kept for. Prior to tremendous data, affiliations would in general store data uncertainly - several terabytes of data won't make high amassing costs; it could even be kept in the live database without causing execution issues. In a customary data setting, there not may even be data reported courses of action set up. Variability: Change in tremendous data's setting implies a few different things. One is the amount of anomalies in the data. These ought to be found by peculiarity and irregularity distinguishing proof methodologies all together for any critical examination to happen. Gigantic data is in like manner factor by virtue of the expansive number of data estimations coming to fruition in light of various diverse data composes and sources. Vacillation can in like manner imply the clashing pace at which gigantic data is stacked into your database.

4 Big Data Technologies/Platform

Brief descriptions of the various Big Data related technologies have been discussed: -

Big data Technology	Description	Supported Platforms
R Studio	R core team have developed, one of open source simulator. Good powerful graphics abilities and functions admirably with Hadoop. Gives a bunch of tools that provides better way of performing estimations, data interpretation and creating charts and graphs.	Mac,Linux, Windows, operating systems and Accomplish complicated data analysis in low cost.
Apache Hadoop (Data-intensive Applications can be processed.	Apache foundation has developed, Used for accumulating, operating, and evaluating the data. It makes clusters of devices and correlate tasks between them.	Windows, Linux, OS X.
Hadoop - YARN (i.e.,Yet Another Resource Negotiators)	Capable to disengage asset the board and preparing parts.	
MapReduce Arch. (Input, Splitting, Mapping, Shuffling, Reducing and Result).	Displayed by Google, structure model and heart of Hadoop, used for parallel computation of tremendous datasets. Model can be accomplished gigantic a huge number of servers inside a cluster of Hadoop.	OS Independent
Hadoop Common	Gives basic Java library methods and utility	
HDFS (Hadoop Distributed File Systems)	A capacity framework for Hadoop, quickly disseminates the data in a few hubs on a bunch. Gives quick execution and dependable copy of data.	Windows, Linux, OS X.
Apache Hadoop Related Projects		
Pig (Pig Latin) [Data flow / Scripting]	Uses literary dialect by Pig Latin in a huge scale detailed examination stage, which creates an arrangement of Map Reduce programs on bunch of Hadoop.	OS Independent
Hbase [From Google BigTable]	In view of HDFS, can store unstructured and semi-sorted out small information. Support section-based database stockpiling (immense table).	OS Independent
Mahout [Library of ML]	Supports different sorts of Data Mining calculations (Batch-based Collaborative separating, Clustering and Classification).	--
Oozie [Java based Web Application.]	Executes on Java Servlet-container-Tomcat. It has workflow manager and Job coordinator.	--
Big top	For packaging and validating the Hadoop ecosystem.	--
Storm [Twitter Product]	Depicted as “Hadoop of real time”, gives real time calculation in distributed way. Profoundly scalable, works strong with programming.	Linux
GridGain	Is is and Alternative to MapReduce which is used in Hadoop. Well suited with HDFS. It gives in-memory handling for snappier assessment.	Windows, Linux, OS X.

HPCC [High Performance Computing Cluster]	Provides superior execution over Hadoop. HPCC is result of LexisNexis Risk arrangements.	Linux
Not Only Structured Query Language (NOSQL) Databases		
Key value Database (DB)	This kind of databases store data in the form of key value pairs. Key may be the primary key and value contains its data. It can be used for storing huge volume of data. It has pliable structure and supports quick transactions. Examples include Redis, Berkeley DB, and Amazon's Dynamo etc.	--
Document store DB	These are used to parse process and store JSON objects. JSON are the light-weight structures. Examples of Document stored DB are CouchDB, MangoDB and Simple DB	--
Column store DB	These kinds of databases require less space than RDBMS because data is stored in columns rather than storing them in rows. Examples include Cassandra and HBase	--
Graph based DB [For IoT it is perfect]	These databases stored and represent data in the form of Nodes and Edges. IoT data can be store in this kind of databases. Examples include Alleograph, FlockDB, and Neo4j etc	--

Examples of NOSQL Database		
MangoDB	It supports document store DB for full index	Windows, Linux, Solaris.
DynamoDB	Amazon's Product	--
Cassandra	It's a Facebook's Product, Maintain by Apache. Used by Reddit, Urban Airship, Twitter and Netflix etc	OS Independent
Neo4j	It is faster than the normal traditional Databases, usually 1000 times. It is a leading. It is a leading a Graph database.	Linux, Windows
CouchDB	Data is put away in JSON structures and can be accessed using JavaScript query or from the web. It is Developed for web.	Windows, Linux, Android.
Hypertable	It's a NoSQL Database developed by Zvents Inc.	Linux, OS X.
Riak	It's Supports key value DB. It offer fault-tolerance, high availability and scalability through distributed NOSQL.	Linux, OS X.
Databases / Data warehouse		
Hive	It is developed by Facebook, it is DataWare for Hadoop cluster and for queries it uses HiveQL.	OS Independent
FlockDB	It is famously called DB of social graphs and it is developed by Twitter.	OS Independent
Hibari	It is used by many telecome companies. It is ordered key-value storage; it ensures reliability and high bandwidth	OS Independent
Data Aggregation and Transfer		
Sqoop	It is utilized to exchange the data between RDBMS and Hadoop. Using this single or multiple tables of SQL databases can be imported to HDFS.	OS Independent
Flume	It is used to moves gigantic amount of log data. It is more flexible and reliable architecture.	Linux

[Apache Product]		

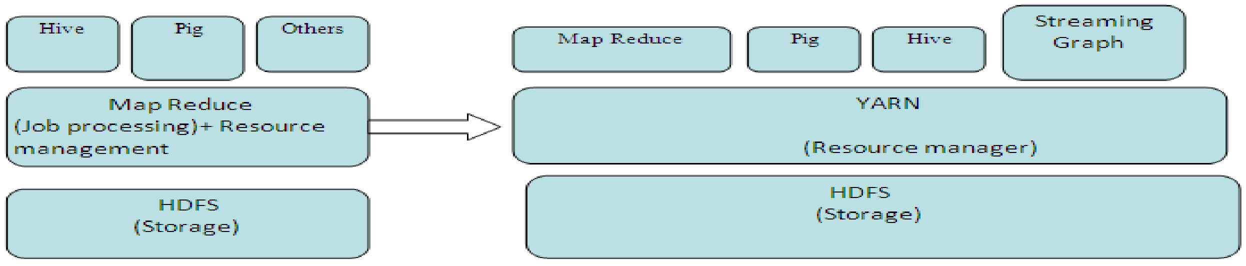
Apache Spark

Spark Environment	❖ Spark gushing – It prepared continuous information, underpins Python, Scala, and Java ❖ Spark MLib: It contains the ML library that contains all learning set of guidelines and diverse parts. ❖ Spark Graph X: It's another diagram API for the parallel chart ❖ Computations: (Spark Resilient Distributed Data sets-RDD-Abstraction) ❖ Spark SQL: It is called as Shark, the latest model and concentrated unit of Spark, which contain SQL and API.
Spark Architecture	It very well may be made both as conveyed or independent venture In the same run time it supports iterative, interactive, batch and streaming computations by authorizing rich applications. The three principle segments are; ❖ Data Storage-HDFS is utilized by Spark ❖ API-Developers to build up an application utilizing API interface(Java, Python, etc) ❖ Management Framework-works in both independent and circulated system ❖ Possibility of implementing both batch processing and stream on a similar framework. ❖ It makes a less intricate improvement, organization and the support of an application.
International Business Machine (IBM) Info Sphere	
IBM InfoSphere	It is a stage which offers a Business class establishment for different enormous information ventures. It is a part of InfoSphere and is an ETL Tool. Versions: MVS edition, Enterprise Edition, Server Edition

4 HADOOP BASIC FRAMEWORK

With the progression of programming building, Hadoop has encircled a domain from four imperative parts that are GFS, MapReduce, BigTable and Chubby. Apache Hadoop is an item structure that sponsorships data genuine scattered applications under a free allow. It enables applications to work with a large number of center points and petabytes of data. Below we are comparing the traditional system and Hadoop.

Parameter	Traditional RDBMS	Hadoop/map reduce
Data Size	Gigabytes	Petabytes/Exabytes
Updates	Interactive and batch	Batch
Transactions	Read and write many times	Write once , read many times
Structure	ACID	None
Integrity	Schema-on-Write	Schema -on-Read
Scaling	Non Linear	Linear
Speed	Slow	Fast



Hadoop 1.x

Hadoop 2.x

Fig2:-Hadoop 1.x vs. Hadoop 2.x

5 SPARK OVERVIEW / FRAMEWORK

The sub project of Hadoop is known as Spark which is developed by Matei Zaharia in AMPLab of UC Berkeley in 2009. Spark was donated to apache software foundation which was open sourced in 2010 with BSD Licensed and this apache project has turned out to be top dimension venture from Feb-2014 and before Spark we have Map-Reduce technology. Hadoop Map-is an item framework for adequately making applications which process tremendous proportions of data (multi-terabyte educational accumulations) in-parallel on huge clusters (a colossal number of centre points) of thing gear in a strong accuse tolerant way. Apache spark is based on Hadoop and this technology is fast cluster computing which designed for fast computation. The Apache Spark includes stream processing and queries and Spark extend Map Reduce model to use it efficiently for many types of computation.

The Apache Spark works on the basis of connected multiples Ram of many computers which are able to process the data in parallel because these data stores on Ram not in spanning disk, this is called cluster computing of in-memory which through it increases the processing speed of computing. The Apache Spark constructed in such way that it is able handle the workload of interactional queries, batch application, streaming, and reiterative algorithms. The Spark supports all workloads as well as it reduces burden of separate tools by management of maintenance.

Problems in Map Reduce:

- Map Reduce is hard to program and needs deliberations
- There is no worked in intuitive mode aside from pig and hive.
- Used for producing cluster reports that assistance in discovering bits of knowledge into recorded inquiries.
- Does not use the memory of the group.
- It is plate arranged totally.
- Writing data pipelines is many-sided and extended

Spark key characteristic:

- Unlimited Scale
- Ease of Development
- In Memory Performance
- Enterprise Platform
- Wide Range of Application
- Combine Workflow

Spark Solution to Map Reduce:

- Difficult—Spark is definitely not hard to program and diverged from MR.
- Spark is Interactive
- Streaming is refined just and besides gives group dealing with and machine learning.
- 100 times speedier than Hadoop.
- In memory enrolling ensures low dormancy and gives putting away limits across finished appropriated experts.
- Simpler and less limited appeared differently in relation to MR.

Spark component

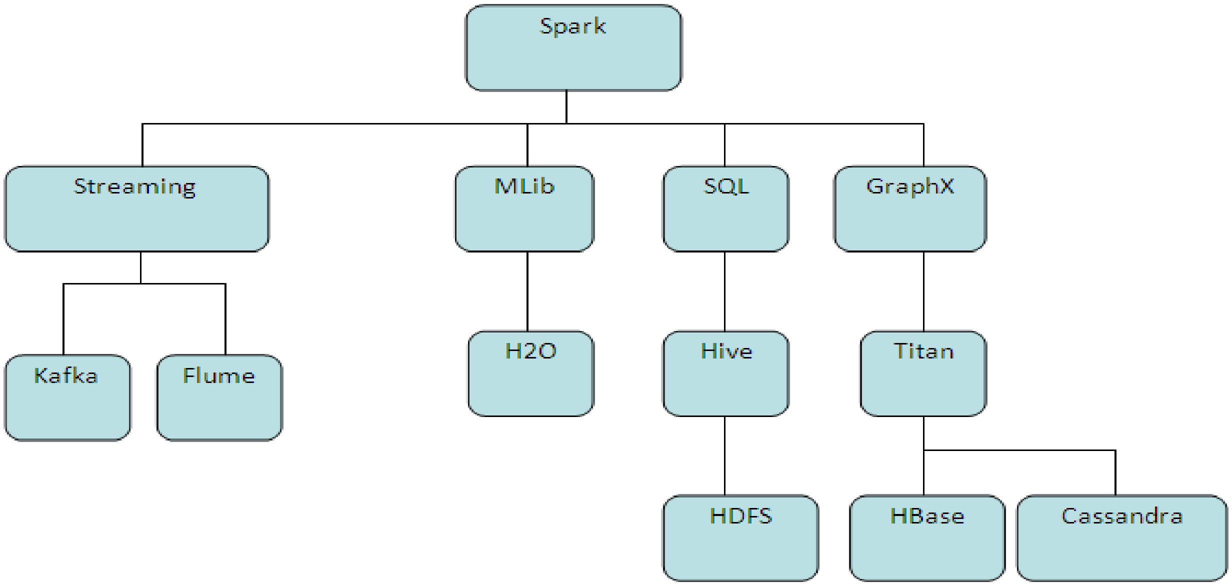


Fig3:- *Spark Component*

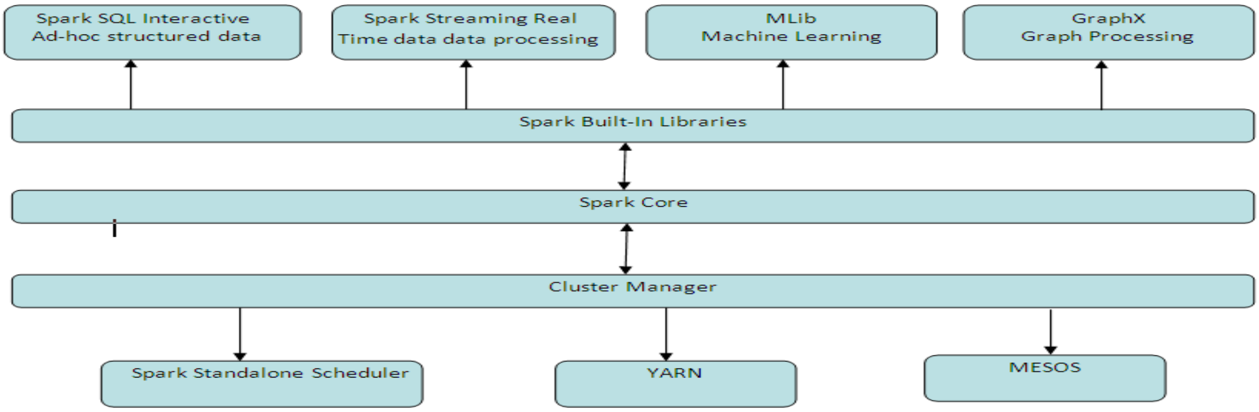


***Fig4:-** Spark Framework/Cluster Manager*

Spark Core
The spark core engine is a distributed execution engine and it is intended to deal with the expansive size of parallel registering and conveyed information handling. The Spark centre architecture design contains many features such as; it provides the API platform to many programming languages for example Python, Java or Scala for application development of distributed ELT, On the top of core many libraries are developed to create the diversion of streaming, machine learning, and SQL workloads, as well as the core, is also responsible to monitor the jobs such as; fault tolerance and scheduling memory management, distribution and with the storage systems of cluster interacting.

Spark Streaming
Spark streaming is useful to the API of center Spark which is utilized to process the continuous streaming data and which is the component of Spark. This Spark stream design is based on the series of RDD (resilient Distributed Dataset) for processing the real time live data and it enables the fault tolerance and high throughput for stream processing of live data

Spark SQL
Remembering the true objective to work with sorted out data of Spark's package which contains Spark SQL[12][13]. HQL (Hive Query Language) is the form of Spark SQL which enables the addressing data through the Apache Hive or SQL and these are the varieties of AQL [14]. The spark SQL joins the Hive table, parquet and JSON by reinforcing the distinctive different sources of data. Besides basically giving a common SQL interface to Spark, Spark SQL in like manner empowers planners to solidify assorted SQL request with the programmed data controls that are reinforced by RDDs in Scala, Python and Java. While this all goes under a solitary application there for it joins SQL with complex examination. This component of solidly consolidating the setting with the rich and drive handling condition gave by Spark enhances it than some other existing open source information stockroom contraption. Sparkle SQL was joined as understanding 1.0 in Spark.

GraphX
The reason to extend the Spark RDD property of graph is that, the Apache Spark has many API in which one of them is known as GraphX which is used for graphs and parallel graphs for computation. The property of Graph is also known as multigraph which may have multiple edges in a parallel way and each and every edge and vertex has the user defined property and the parallel edges have the multiple relationship between the same vertices. By attaching the directed multigraph properties to each and every vertex and edge by using Resilient Distribution property of Graph, the component GraphX of Spark extends the Spark RDD abstraction at a very high level.
GraphX contains the collection of builders and graph algorithms which is growing for simplifying graph analytics tasks as well as it optimizes various Pregel APIs and shows a set of operators such as map-reduce triplets, sub-graph and join vertices for supporting the graph computation.

MILib (Machine Learning)
MLlib points for Machine Learning Library. Performing machine learning in Apache Spark MILib is used. Machine learning can be implemented by using R language and Python also which provides better visibility and graphical representation

Here we have compared the Apache Hadoop and Spark Technologies: -

Features	Apache Hadoop	Apache Spark
Data Processing Engine	At the core Hadoop's Map Reduce is batch processing Engine	At the core Apache Spark is batch processing Engine
Language Support	Java, but other languages such as C++, C, Ruby, Groovy, Python and Perl also supported streaming Hadoop.	Supports python, Java, R and Scala.
Language Developed	Hadoop is developed in Java	Spark is developed in Scala

Processing Speed	Map-Reduce processes data much laggy than Spark and Flink.	Spark processes 100 times faster than Map-Reduce, because of it is in-memory processing system.
Iterative Processing	Does not support iterative processing natively.	Spark in slots iterates its data. For in Spark, iterative processing, every iteration must be planned and performed separately
Stream Processing	Mapreduce is purely batch-oriented data processing tool. It doesn't support stream processing	Spark Uses micro-batches for all workload streaming. However, it's not sufficient for cases where large live streams need to be processed data and results with low latency in real time
Computation Model	Map Reduce batch-oriented model adopted. Batch processes data essentially at rest, taking a large amount Processing the data at once and then writing the output.	Spark's core also follows batch model but has adopted micro-batching. Micro-batches are an essentially used for handling near real-time processing data model.
Memory Management	Hadoop provides configurable Memory management. Admin can configure it using configurations files.	Spark provides configurable memory management, although with the latest release of Spark 1.6, automating memory management is also included in Spark as well.
Windows criteria	NA	Spark has time-based Window criteria.
Optimization	In Map Reduce jobs has to be manually optimized.	In Apache Spark jobs has to be manually optimized.
Latency	Apache Hadoop has greater latency than Flink and spark.	Apache Spark has high latency as compared to Apache Flink.
Fault tolerance	MapReduce is highly fault tolerant, from scratch no need to restart the application if there is a failure.	Spark Streaming recovers lost work and delivers exactly once out of the box without additional code or setup
Performance	Hadoop's performance is slower than Spark and Flink	Though Apache Spark has a wonderful community background and now it is contemplating as fully fledged community. But It's stream processing is not as capable as Apache Flink because it uses micro-batch processing.
Duplicate elimination	NA	Spark process every records exactly once hence eliminates duplication.
Compatibility	Map Reduce and Spark are compatible to each other.	Spark and MapReduce are compatible to each other and Spark shares all MapReduce's compatibility for , file formats, data sources and business intelligence tools via ODBC and JDBC.
Security	Hadoop helps Kerberos Authentication, that's a bit painful to handle. Supports HDFS Access control lists(ACLs) and a conventional model for file permissions. Third- party vendors have, however, allowed companies to leverage Kerberos Active Directory and LDAP for authentication.	The security of Spark is a bit sparse at the moment, only Supporting shared secret authentication(password authentication). The safety funny thing is that if you run Spark on Hadoop, it uses HDFS ACLs and permissions on file level. Spark can also run on YARN gives us the ability.
Iterative Data Flow	Map Computer data flow reduction has no loops, it is a chain of stages; at each stage you progress with the output of the previous stage and the input for the next stage.	Although ML is a cyclic data flow, it is represented in the spark as a direct acyclic graph
Visualization	All the BI tools like JasperSoft, SAP Business Objects, Qlikview, Tableu, Zoom Data, etc. have provided connectivity with hadoop & its ecosystem.	All the BI tools like JasperSoft, SAP Business Objects, Qlikview, Tableu, Zoom Data, etc. have provided connectivity with Spark. Spark can also be integrated to Apache. It provides analysis of data, Discovery, visualization and collaboration as well as ingestion.
Data size	Exabytes	Zettabyte/Yottabyte

6 CONCLUSION

This paper provides the comparison and performance of various technologies used in apache spark and hadoop. We have discussed the various works done by several authors on spark and Hadoop performance. Spark and related technologies is much better than traditional databases for processing perspective. Here we discussed variety of data available how huge amount of data processed and analyzed. Here we have mentioned the feature and components of spark, Hadoop and Bigdata related technologies.

REFERENCES

[1] K.-H. Lee, Y.-J. Lee, H. Choi, Y. D. Chung, and B. Moon, "Paralleldata processing with mapreduce: a survey," SIGMOD Record, vol. 40, no. 4, pp. 11–20, 2011.

[2] S. Sakr, A. Liu, and A. G. Fayoumi, "The family of mapreduce and large-scale data processing systems," ACM Comput. Surv., vol. 46, no. 1, p. 11, 2013.

[3] Y. He, R. Lee, Y. Huai, Z. Shao, N. Jain, X. Zhang, and Z. Xu, "Rcfile:A fast and space-efficient data placement structure in mapreduce-based warehouse systems," in ICDE, 2011, pp. 1199–1208.

[4] "Hive," http://hive.apache.org.

[5] A. Pavlo, E. Paulson, A. Rasin, D. J. Abadi, D. J. DeWitt, S. Madden, and M. Stonebraker, "A comparison of approaches to large-scale data analysis," in SIGMOD Conference, 2009, pp. 165–178.

[6] A. Floratou, N. Teletia, D. J. DeWitt, J. M. Patel, and D. Zhang, "Can the elephants handle the nosql onslaught?" PVLDB, vol. 5, no. 12, pp. 1712–1723, 2012

[7] A. Floratou, U. F. Minhas, and F. Ozcan,"Sql-on-Hadoop: Full circle back to shared-nothing database architectures," PVLDB, vol. 7, no. 12, pp. 1295–1306, 2014.

[8] Yongqiang He , Rubao Lee , Yin Huai , Zheng Shao , Jain, N ,Xiaodong Zhang , Zhiwei Xu,"RCFile: A fast and space-efficient data placement structure in MapReduce-based warehouse systems" Data Engineering (ICDE), 2011 IEEE 27th International Conference on DOI: 10.1109/ICDE.2011.5767933 Publication Year: 2011 , Page(s): 1199- 1208.

[9] Taoying Liu,Jing Liu,Hong Liu,Wei Li,"A performance evaluation of Hive for scientific data management", Big Data, 2013 IEEE International Conference on DOI: 10.1109/BigData.2013.6691696 Publication Year: 2013 , Page(s): 39 – 46.

[10] Jingmin Li,"Design of real-time data analysis system based on Impala", Advanced Research and Technology in Industry Applications (WARTIA),2014 IEEE Workshop on DOI: 10.1109/WARTIA.2014.6976427 Publication Year: 2014 , Page(s):934 – 936.

[11]Cloudera http://www.cloudera.com/content/cloudera/en/home.html

[12] Michael Armbrusty, Reynold S. Xiny, Cheng Liany, Yin Huaiy,Davies Liuy, Joseph K. Bradleyy,Xiangrui Mengy, TomerKaftanz, Michael J. Franklinyz, Ali Ghodsiy, Matei Zahariay, "Spark SQL: Relational Data Processing in Spark", AMPLab, UC Berkeley, 2015

[13] Zhijie Han, Yujie Zhang, "Spark:A Big Data ProcessingPlatform Based On Memory Computing", IEEE 2015 SeventhInternational Symposium on Parallel Architectures, Algorithms and Programming,12-14 Dec. 2015.

[14] (Accessed on 10th March 2016) Hive Querry Language [Online] Available: https://docs.treasuredata.com/articles/hive.

[15] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," in OSDI, 2004, pp. 137–150.

[16] https://techjury.net/stats-about/big-data-statistics